

## Portfolio 7

Morten Street, Sigrid Snapfield, Anders Wheelman & Gustav Helmet

3/24/2020 – due 31<sup>st</sup> of March

```
#Setup
library(pacman)
pacman::p_load("tidyverse", "pastecs", "lmerTest", "nlme", "ggpubr", "emmeans")

face_exp_2016<- read.csv("face_exp_data_all_160310.csv", sep=";")
face_exp_2017<- read.csv("face_exp_all_logs_2017.csv", sep=";")
#Binding the two datasets together
face_exp<-rbind(face_exp_2016,face_exp_2017)
#conditions are coded in the "cond_blue", "cond_emo" and "freq" variables

#Making a vector with the collumns that needs to be transformed to factors
collums <- c("cond_emo","cond_blue","no","duration_frames","year")
#Using lapply to set all the selected collums as factors
face_exp[collums] <- lapply(face_exp[collums], factor)
```

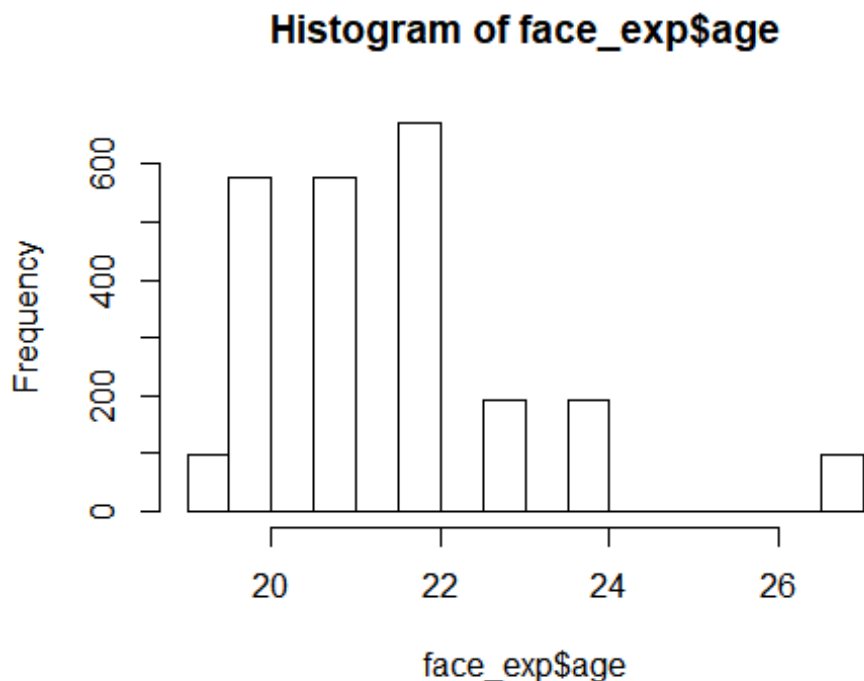
### 1. Understanding the experiment

**1.a.Comprehension question. Please explain which factor was between-participants and which were within participants and why.**

- The Cond\_emo (emotion) and the Cond\_blue (color) are within participants factors since all participants were exposed to all conditions in the factors.
- The freq factor (frequency color manipulation group) is a between participant factor since participants were only exposed to one frequency manipulation.

**1.b. What was the age range of the participants?**

```
hist(face_exp$age)
```



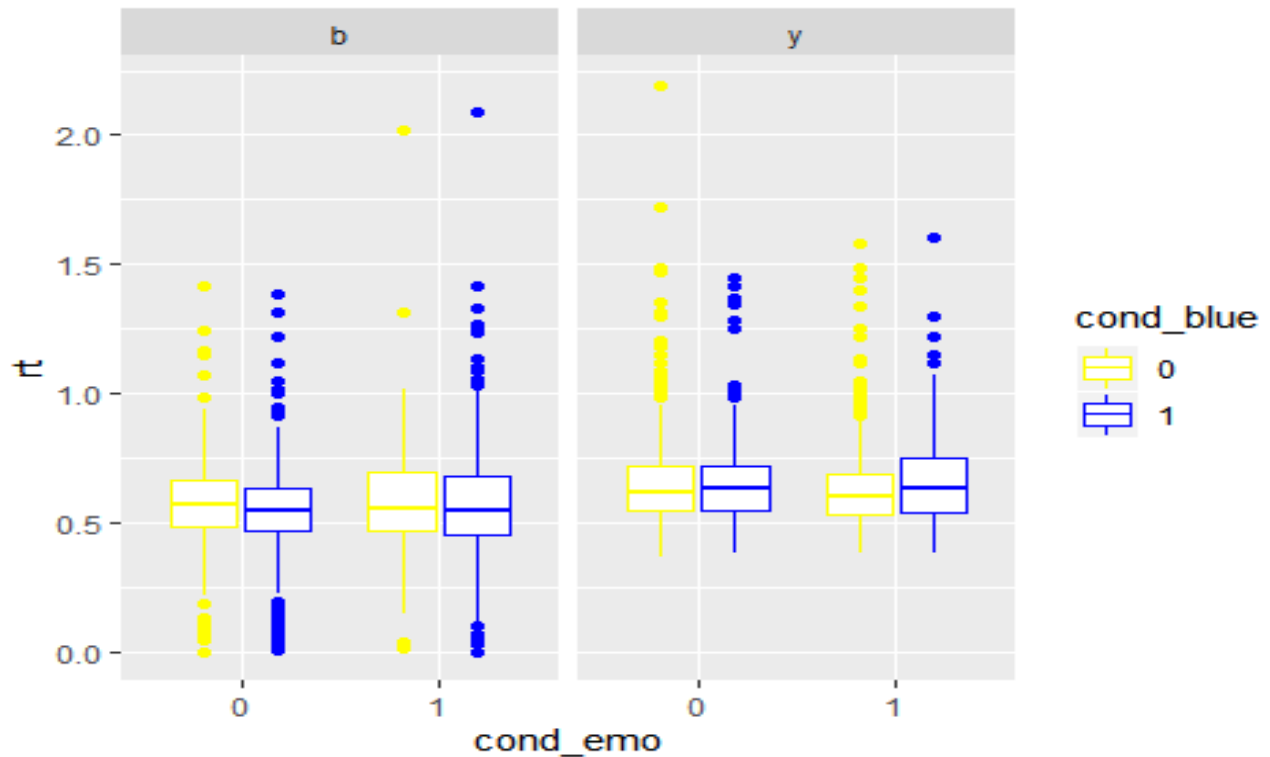
```
min(face_exp$age)
## [1] 19
max(face_exp$age)
## [1] 27
```

The age ranged between 19 and 27

## 2. Data exploring and preparation

**2.a:** make a box-plot of the data with RT on the y-axis and emotional condition on the x-axis. Make a box-plot for each of the color conditions by using "fill". Use `facet_wrap()` to make two separate graphs for each frequency group. Give the boxes colors that matches the stimuli, eg. use " + `scale_fill_manual(values=c("yellow","blue","yellow","blue","yellow","blue","yellow","blue"))`".

```
#The big one
ggplot(face_exp, aes(cond_emo, rt, colour = cond_blue))+
  geom_boxplot()+
  facet_wrap(~freq)+
  scale_colour_manual(values=c("yellow","blue","yellow","blue","yellow","blue",
    "yellow","blue"))
## Warning: Removed 3 rows containing non-finite values (stat_boxplot).
```



**2.b: Comprehension question. Explain why this plot shows that there is something wrong with the data.**

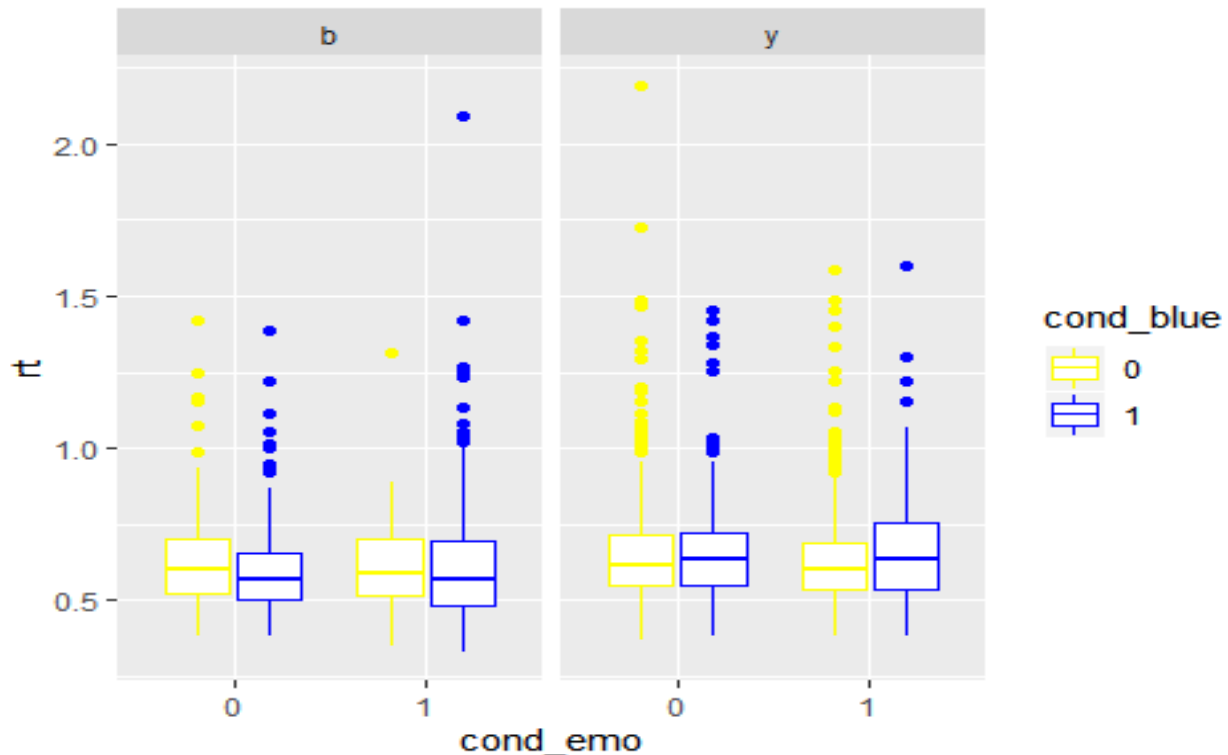
There seems to be a lower limit of RT in the yellow frequency group, since the lower whiskers of the four boxplots in this group have the same length. Besides from that, the boxplots in the blue frequency group also have a problem, because the whiskers seem to extend to datapoints with a RT very, very close to 0 seconds, which is probably due to error. Though it is curious that it is only in the blue freq group that this error happens.

**2.c.: Make a subset of the data, including only correct responses.**

```
#Making a subset only with correct answers
df <- subset(face_exp, face_exp$correct_resp == 1)
```

**2.d.: Make another boxplot similar to that in 2.a. Did it solve the observed problem?**

```
ggplot(df, aes(cond_emo, rt, colour = cond_blue))+
  geom_boxplot()+
  facet_wrap(df$freq)+
  scale_colour_manual(values=c("yellow","blue","yellow","blue","yellow","blue",
    "yellow","blue"))
```



Yes, it solved the problem with *rt* approaching 0 in the blue frequency group. By eyeballing the data, it still seems like there is a lower limit for the yellow frequency group condition.

**2.e.: Use the `by()` function and `stat.desc` (in `library(pastecs)`) to get descriptive measures for the different conditions (e.g. see Field's book chapter 5.5.3.2.). Try to investigate the three hypotheses based on the descriptive statistics - would you expect any of the statistical analyses to be significant based on the descriptive stats?**

```
#Hypothesis 1: The cond_blue factor
by(df$rt, df[, 6], stat.desc, basic = F, options(scipen = 100))

## df[, 6]: 0
##      nbr.val      nbr.null      nbr.na      median      mean
## 1162.0000000    0.0000000    0.0000000    0.6021989    0.6389946
##      std.dev
##      0.1716556
## -----
## df[, 6]: 1
##      nbr.val      nbr.null      nbr.na      median      mean
## 1043.0000000    0.0000000    0.0000000    0.6009792    0.6246180
##      std.dev
##      0.1679895
```

H1: The index finger (blue) trials will lead to a shorter response time than middle finger (yellow) trials.

Looks like the means of the two groups are practically the same, very little difference. We would not expect this to be significant.

*#Hypothesis 2: The cond\_emo factor*

```
by(df$rt, df[, 5], stat.desc, basic = F, options(scipen = 100))
```

```
## df[, 5]: 0
##      median      mean      SE.mean CI.mean.0.95      var
## 0.602302368 0.635330286 0.005109437 0.010025273 0.028925827
##      std.dev      coef.var
## 0.170075945 0.267696895
## -----
## df[, 5]: 1
##      median      mean      SE.mean CI.mean.0.95      var
## 0.600937149 0.629026784 0.005133648 0.010072889 0.028910715
##      std.dev      coef.var
## 0.170031513 0.270308859
```

H2: Fearful faces will yield a shorter response time than neutral faces.

Again, there is a very little difference in the means at around 0.0063 seconds, which is not alot. We would not expect this to be significant.

*#Hypothesis 3: The freq factor*

```
by(df$rt, list(df$cond_blue, df$freq), stat.desc, basic = F, options(scipen = 100))
```

```
## : 0
## : b
##      median      mean      SE.mean CI.mean.0.95      var
## 0.586712092 0.621161189 0.009198697 0.018107382 0.023777104
##      std.dev      coef.var
## 0.154198262 0.248241946
## -----
## : 1
## : b
##      median      mean      SE.mean CI.mean.0.95      var
## 0.568222113 0.601009045 0.006671442 0.013102040 0.026927423
##      std.dev      coef.var
## 0.164095773 0.273033782
## -----
## : 0
## : y
##      median      mean      SE.mean CI.mean.0.95      var
## 0.602794179 0.644682665 0.005948624 0.011675146 0.031175177
##      std.dev      coef.var
## 0.176564936 0.273878834
## -----
## : 1
## : y
##      median      mean      SE.mean CI.mean.0.95      var
```

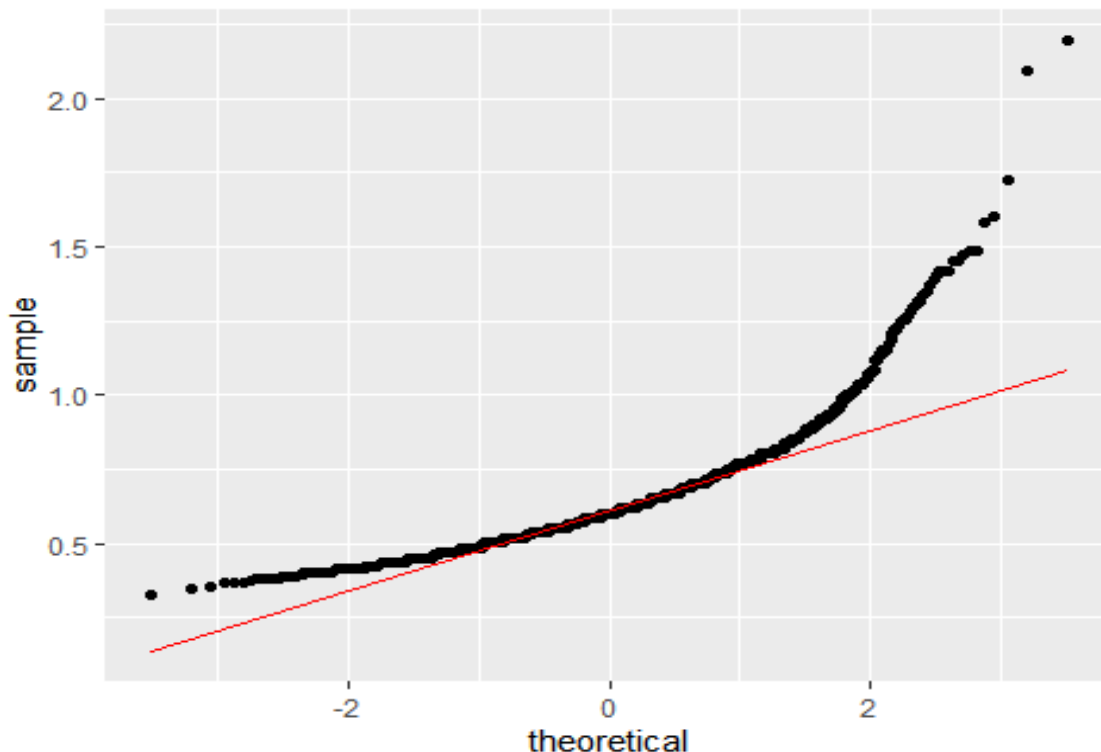
```
## 0.634669934 0.657228642 0.008028866 0.015779992 0.028234659
##      std.dev      coef.var
## 0.168031719 0.255667068
```

H3: Infrequent stimuli will yield longer responses time than frequent. This should surface as an interaction between color and frequency group.

By comparing the means of blue/yellow stimuli in the blue frequency manipulation group, we find a small advantage for blue stimuli. The same pattern is seen for yellow stimuli in the yellow group. However the standard deviations are relatively large; approximately 1/3 of the mean, so we wouldn't expect these interactions to be significant.

## 2.f.: Explore if the RT data is normally distributed using a qq-plot (e.g. `qqnorm()`).

```
ggplot(df, aes(sample = df$rt))+
  stat_qq()+
  stat_qq_line(colour = "red")
```



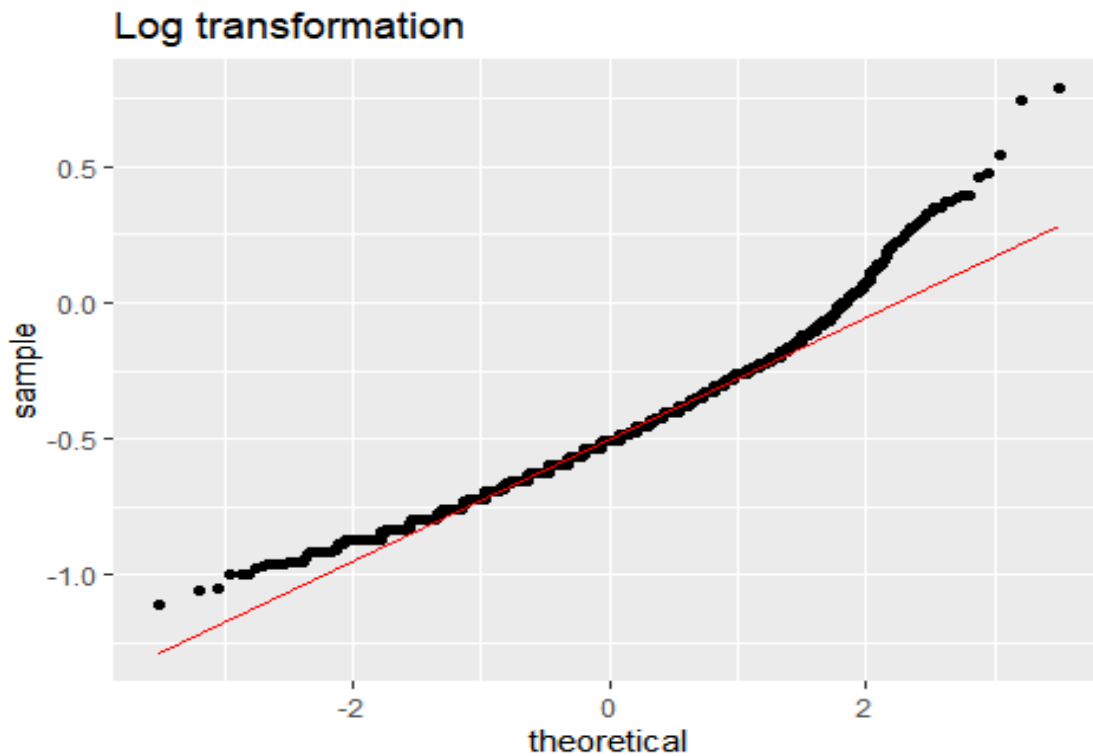
By visually inspecting the data, the `rt` variable does not seem to be normally distributed.

## 2.g.: log-transform the RT data.

```
#Making a log transformation
df <- df %>% mutate(RT_log=log(df$rt))
```

2.h.: Use a qq-plot to explore if the transformed data appear more normal than the untransformed.

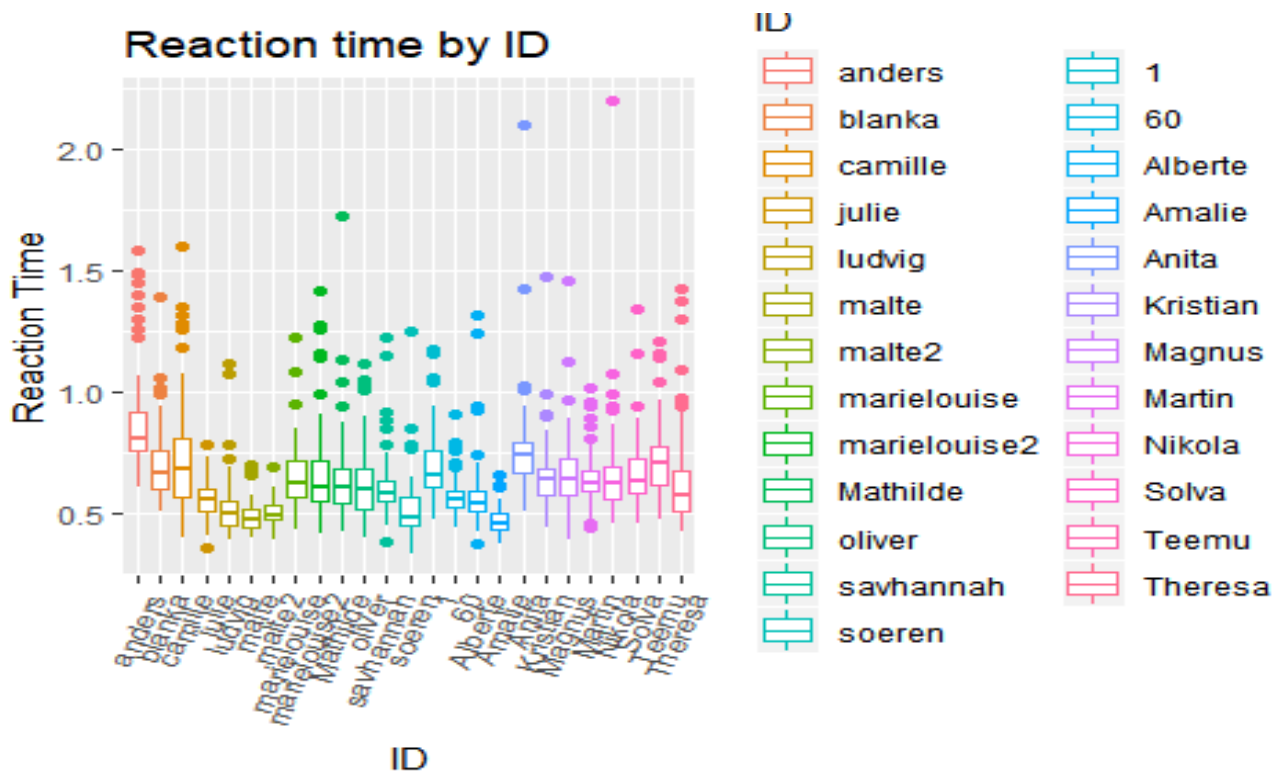
```
ggplot(df, aes(sample = df$RT_log))+  
  stat_qq()+  
  stat_qq_line(colour = "red")+  
  ggtitle("Log transformation")
```



After log transforming the rt variable, it seems to approach normal distribution.

2.i.: Make a plot that explores the response times for participants, individually, using a box-plot. Does anybody stick out as unusual?

```
ggplot(df, aes(ID,rt, colour = ID))+  
  geom_boxplot()+  
  theme(axis.text.x = element_text(angle = 70, hjust = 1))+  
  labs(title = "Reaction time by ID", y= "Reaction Time")
```



We see a variance of reaction times across individuals, but the variance is not high enough to cause concern.

### 3. Data analysis

**3.a Make mixed effects model where you predict reaction time using the three factors as fixed effects, and include random intercepts for each participant (use "ID" from the log). Include 2-way and 3-way interactions as well. To do this use lme() from the "nlme" package, and use maximum-likelihood as estimation method(method = "ML").**

*#We only use the correct results and make a full model*

*#Full model*

```
m1 <- lme(rt ~ cond_blue + cond_emo + freq + cond_blue*freq*cond_emo, random
= ~1|ID, data = df, method = "ML", na.action = na.omit)
```

**3.b.: Report the t-statistics using summary().**

```
summary(m1)
```

```
## Linear mixed-effects model fit by maximum likelihood
## Data: df
##          AIC          BIC      logLik
##   -2137.346  -2080.361  1078.673
##
## Random effects:
```



```
## Formula: ~1 | ID
##          (Intercept) Residual
## StdDev:   0.0843167 0.1455144
##
## Fixed effects: rt ~ cond_blue + cond_emo + freq + cond_blue * freq * cond_emo
##
##              Value Std.Error   DF   t-value p-value
## (Intercept)   0.6225921 0.02833777 2174 21.970398  0.0000
## cond_blue1    -0.0414825 0.01492333 2174 -2.779709  0.0055
## cond_emo1     -0.0256658 0.01740680 2174 -1.474468  0.1405
## freq          0.0320796 0.03688862   23  0.869633  0.3935
## cond_blue1:freq 0.0430997 0.01915627 2174  2.249902  0.0246
## cond_blue1:cond_emo1 0.0455863 0.02106492 2174  2.164084  0.0306
## cond_emo1:freq  0.0040625 0.01998750 2174  0.203250  0.8390
## cond_blue1:cond_emo1:freq -0.0232987 0.02709941 2174 -0.859749  0.3900
## Correlation:
##              (Intr) cnd_b1 cnd_m1 freq  cnd_b1: cn_1:_1
## cond_blue1    -0.363
## cond_emo1     -0.311  0.589
## freq         -0.768  0.279  0.239
## cond_blue1:freq  0.283 -0.779 -0.459 -0.285
## cond_blue1:cond_emo1 0.256 -0.707 -0.826 -0.197  0.551
## cond_emo1:freq  0.271 -0.513 -0.871 -0.273  0.525  0.719
## cond_blue1:cond_emo1:freq -0.199  0.550  0.642  0.201 -0.706 -0.777
##              cnd_m1:
## cond_blue1
## cond_emo1
## freq
## cond_blue1:freq
## cond_blue1:cond_emo1
## cond_emo1:freq
## cond_blue1:cond_emo1:freq -0.737
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -2.1325397 -0.5515993 -0.1511939  0.3101766 10.5555973
##
## Number of Observations: 2205
## Number of Groups: 25
```

There is one significant variable (cond\_blue: t-value(2174) = -2.7797, p<.05) and two significant interactions (cond\_blue1:cond\_emo1: t-value(2174) = 2.1641, p<.05 and cond\_blue1:freq: t-value(2174) = 2.2499, p<.05)

### 3.c.: Report the F-statistics using anova() and type='sequential', which gives you type='I' analysis.

```
anova(m1, type = "sequential") #Type I

##              numDF denDF   F-value p-value
## (Intercept)      1  2174 1325.8126  <.0001
```

```
## cond_blue          1  2174    0.0029  0.9570
## cond_emo           1  2174    1.0161  0.3136
## freq               1    23    2.1921  0.1523
## cond_blue:freq      1  2174    5.3942  0.0203
## cond_blue:cond_emo  1  2174    7.6476  0.0057
## cond_emo:freq       1  2174    0.4062  0.5240
## cond_blue:cond_emo:freq 1  2174    0.7392  0.3900
```

Two interactions are found to have a significant effect: `cond_blue1:cond_emo1`: F-value(2174) = 7.63,  $p < 0.01$  and `cond_blue1:freq`: F-value(2174) = 5.41,  $p < 0.05$ . When comparing with the T-statistics from 3.b we now see that `cond_blue` as a main effect is non-significant.

### 3.d.: Report the F-statistics using `anova()` and `type='marginal'`. Why might there be differences between results from 3.c and 3.d?

```
anova(m1, type = "marginal") #Type III

##              numDF denDF  F-value p-value
## (Intercept)      1  2174 482.6984  <.0001
## cond_blue        1  2174   7.7268  0.0055
## cond_emo         1  2174   2.1741  0.1405
## freq             1    23   0.7563  0.3935
## cond_blue:freq    1  2174   5.0621  0.0246
## cond_blue:cond_emo 1  2174   4.6833  0.0306
## cond_emo:freq     1  2174   0.0413  0.8390
## cond_blue:cond_emo:freq 1  2174   0.7392  0.3900
```

When using 'sequential' we get type I sum of squares. When using 'marginal' we get type III sum of squares.

Using type III sums of squares retains all of the significant values whereas using sequential sums of squares makes `cond_blue` nonsignificant. The type III sums of squares is the most appropriate in this case, because type I requires the variables to be independent (which our are not, since it is a mixed model), plus the order of the predictors matter, which we don't want in this case.

### 3.e.: Make a new model including a random slope from trial number ('no' in the log-file). Repeat 3.b. What does the inclusion of such a random slope model?

#### Did it change the results?

```
m2 <- lme(rt ~ cond_blue + cond_emo + freq + cond_blue*cond_emo*freq, random
= ~no|ID, data = df, method = "ML")
```

```
summary(m2)
```

```
## Linear mixed-effects model fit by maximum likelihood
## Data: face_exp_cor
##           AIC           BIC logLik
##   -2155.601  -2087.219  1089.8
##
```

```

## Random effects:
## Formula: ~no | ID
## Structure: General positive-definite, Log-Cholesky parametrization
##           StdDev      Corr
## (Intercept) 0.0854302431 (Intr)
## no          0.0007812995 -0.246
## Residual    0.1438851695
##
## Fixed effects: rt ~ cond_blue + cond_emo + freq + cond_blue:cond_emo + con
d_blue:freq +      cond_emo:freq + cond_emo:freq:cond_blue
##                               Value Std.Error   DF   t-value p-value
## (Intercept)          0.6166732 0.02805559 2174 21.980399 0.0000
## cond_blue1          -0.0408069 0.01479698 2174 -2.757786 0.0059
## cond_emo1           -0.0257103 0.01725856 2174 -1.489714 0.1364
## freq               0.0394512 0.03649854   23  1.080898 0.2909
## cond_blue1:cond_emo1  0.0473595 0.02087944 2174  2.268236 0.0234
## cond_blue1:freq      0.0419984 0.01900181 2174  2.210231 0.0272
## cond_emo1:freq       0.0034605 0.01982423 2174  0.174557 0.8614
## cond_blue1:cond_emo1:freq -0.0261979 0.02688034 2174 -0.974613 0.3299
## Correlation:
##                               (Intr) cnd_b1 cnd_m1 freq  cn_1:_1 cnd_b1: cnd
_m1:
## cond_blue1          -0.366
## cond_emo1           -0.310  0.587
## freq               -0.769  0.282  0.238
## cond_blue1:cond_emo1  0.258 -0.706 -0.825 -0.198
## cond_blue1:freq      0.285 -0.779 -0.457 -0.288  0.550
## cond_emo1:freq       0.270 -0.511 -0.871 -0.273  0.718  0.524
## cond_blue1:cond_emo1:freq -0.200  0.548  0.641  0.203 -0.777 -0.705 -0.
736
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -2.0061538 -0.5565794 -0.1530110  0.3038404 10.4244362
##
## Number of Observations: 2205
## Number of Groups: 25

```

It models that every participant responds differently to changes in no, i.e. trial number. The factors that were significant in the last model are still significant. The results have only been changed marginally. The significant t-statistics are as follows:

The main effect of colour (cond\_blue) is significant,  $t(2174) = -2.76$ ,  $p < .01$  ( $p = 0.0059$ ).

The interaction between colour and emotion is significant,  $t(2174) = 2.27$ ,  $p < .05$  ( $p = 0.0234$ ).

The interaction between colour and frequency group is also significant,  $t(2174) = 2.21$ ,  $p < .05$  ( $p = 0.0272$ ).

**3.f.: Make a model comparison of model 3.a and 3.e using anova(). Did the inclusion of a random slope significantly improve the model?**

```
anova(m1, m2)
```

```
##      Model df      AIC      BIC   logLik   Test  L.Ratio p-value
## m1      1 10 -2137.346 -2080.361 1078.673
## m2      2 12 -2155.601 -2087.219 1089.800 1 vs 2 22.25427 <.0001
```

The inclusion of a random slope did significantly improve the model, lratio = 22.25, p<.0001.

**3.g.: Response times are correlated in time which goes against the assumption of independence. It might therefore be an idea to model this by including a so-called auto-regressive component in the model (e.g. this is default in SPM analyses of fMRI-data). In lme(), this is done by adding the following to the model specification: “cor=corAR1(form=~1|ID)”. Make a new model comparison. Does that have an effect?**

```
# creating the new model
```

```
m3 <- lme(rt ~ cond_blue + cond_emo + freq + cond_blue*cond_emo*freq, random
= ~no|ID, data = df, method = "ML", na.action = na.omit, cor=corAR1(form=~1|ID))
```

```
summary(m3)
```

```
## Linear mixed-effects model fit by maximum likelihood
## Data: face_exp_cor
##      AIC      BIC   logLik
## -2178.676 -2104.595 1102.338
##
## Random effects:
## Formula: ~no | ID
## Structure: General positive-definite, Log-Cholesky parametrization
##           StdDev      Corr
## (Intercept) 0.0849782729 (Intr)
## no          0.0007389268 -0.232
## Residual    0.1445441591
##
## Correlation Structure: AR(1)
## Formula: ~1 | ID
## Parameter estimate(s):
##      Phi
## 0.1234472
## Fixed effects: rt ~ cond_blue + cond_emo + freq + cond_blue * cond_emo * f
req
##                               Value Std.Error   DF   t-value p-value
```

```

## (Intercept)          0.6182015 0.02807205 2174 22.021959 0.0000
## cond_blue1          -0.0398698 0.01453090 2174 -2.743791 0.0061
## cond_emo1           -0.0275362 0.01697363 2174 -1.622294 0.1049
## freqy               0.0415594 0.03655247 23 1.136980 0.2673
## cond_blue1:cond_emo1 0.0487635 0.02052286 2174 2.376058 0.0176
## cond_blue1:freqy     0.0367991 0.01871400 2174 1.966393 0.0494
## cond_emo1:freqy      0.0025412 0.01951340 2174 0.130230 0.8964
## cond_blue1:cond_emo1:freqy -0.0245981 0.02642545 2174 -0.930849 0.3520
## Correlation:
##                      (Intr) cnd_b1 cnd_m1 freqy  cn_1:_1 cnd_b1: cnd
_m1:
## cond_blue1          -0.358
## cond_emo1           -0.304  0.583
## freqy              -0.768  0.275  0.234
## cond_blue1:cond_emo1 0.252 -0.706 -0.822 -0.193
## cond_blue1:freqy     0.278 -0.776 -0.453 -0.281  0.548
## cond_emo1:freqy      0.265 -0.507 -0.870 -0.268  0.715  0.521
## cond_blue1:cond_emo1:freqy -0.196  0.548  0.638  0.199 -0.777 -0.706 -0.
734
##
## Standardized Within-Group Residuals:
##           Min           Q1           Med           Q3           Max
## -2.0185321 -0.5603134 -0.1591011  0.3036637 10.3505386
##
## Number of Observations: 2205
## Number of Groups: 25

anova(m1,m3, type = "marginal")

##      Model df      AIC      BIC  logLik  Test  L.Ratio p-value
## m1      1 10 -2137.346 -2080.361 1078.673
## m3      2 13 -2178.676 -2104.595 1102.338 1 vs 2 47.32931 <.0001

```

Adding the auto-regressive component significantly improves the model,  $l.ratio = 47.33$ ,  $p < .0001$ .

## 4. Results and interpretation.

**4.a.: Comprehension question.** If you were to report these results, which model would you use and why? Below are some ideas that you may want to consider:

*# In addition to the full model, we make three new models varying in complexity.*

```

#Null model
m0 <- lme(rtn~1, data = df, random = ~1|ID, method = "ML")
summary(m0)

```

```

## Linear mixed-effects model fit by maximum likelihood
## Data: df
##      AIC      BIC    logLik
## -2134.03 -2116.934 1070.015
##
## Random effects:
## Formula: ~1 | ID
##      (Intercept)  Residual
## StdDev:  0.08807805 0.1460226
##
## Fixed effects: rt ~ 1
##              Value Std.Error   DF  t-value p-value
## (Intercept) 0.6256168 0.01789727 2180 34.95599      0
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -2.1936311 -0.5495984 -0.1497545  0.3118043 10.5684232
##
## Number of Observations: 2205
## Number of Groups: 25

# Model not including interactions
m1_simple <- lme(rt ~ cond_blue + cond_emo + freq, random = ~1|ID, data = df,
method = "ML", na.action = na.omit)
summary(m1_simple)

## Linear mixed-effects model fit by maximum likelihood
## Data: df
##      AIC      BIC    logLik
## -2131.153 -2096.963 1071.577
##
## Random effects:
## Formula: ~1 | ID
##      (Intercept)  Residual
## StdDev:  0.08437187 0.1459877
##
## Fixed effects: rt ~ cond_blue + cond_emo + freq
##              Value Std.Error   DF  t-value p-value
## (Intercept)  0.5997733 0.02652597 2178 22.610798  0.0000
## cond_blue1   0.0002616 0.00664599 2178  0.039366  0.9686
## cond_emo1    -0.0062578 0.00622875 2178 -1.004663  0.3152
## freq         0.0513837 0.03470116   23  1.480750  0.1522
## Correlation:
##      (Intr) cnd_b1 cnd_m1
## cond_blue1 -0.172
## cond_emo1  -0.118  0.007
## freq       -0.743  0.068  0.001
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max

```

```
## -2.1751660 -0.5532182 -0.1575970 0.2961337 10.5456513
##
## Number of Observations: 2205
## Number of Groups: 25

# 2-way model - this models our hypotheses
m1_2way <- lme(rt ~ cond_blue + cond_emo + freq + cond_blue*freq, random = ~1
|ID, data = df, method = "ML", na.action = na.omit)
summary(m1_2way)

## Linear mixed-effects model fit by maximum likelihood
## Data: df
##      AIC      BIC    logLik
##   -2134.539 -2094.65 1074.269
##
## Random effects:
## Formula: ~1 | ID
##      (Intercept) Residual
## StdDev: 0.08434621 0.145808
##
## Fixed effects: rt ~ cond_blue + cond_emo + freq + cond_blue * freq
##              Value Std.Error   DF  t-value p-value
## (Intercept)  0.6128590 0.02711547 2177 22.601821 0.0000
## cond_blue1   -0.0188006 0.01056517 2177 -1.779489 0.0753
## cond_emo1    -0.0063265 0.00622257 2177 -1.016708 0.3094
## freq         0.0342012 0.03547932   23  0.963975 0.3451
## cond_blue1:freq 0.0315026 0.01358186 2177  2.319460 0.0205
## Correlation:
##              (Intr) cnd_b1 cnd_m1 freq
## cond_blue1   -0.268
## cond_emo1    -0.117  0.008
## freq         -0.754  0.204  0.002
## cond_blue1:freq 0.208 -0.778 -0.005 -0.209
##
## Standardized Within-Group Residuals:
##      Min      Q1      Med      Q3      Max
## -2.1492017 -0.5498854 -0.1706824  0.2916462 10.5871319
##
## Number of Observations: 2205
## Number of Groups: 25
```

Rule number 1: Report the first model you did. The first model is the fully complex model, taking into account all interactions. There is one significant variable (cond\_blue: T-value(2174) = -2.7797,  $p < 0.05$ ) and two significant interactions (cond\_blue1:cond\_emo1: T-value(2174) = 2.1641,  $p < 0.05$  and cond\_blue1:freq: T-value(2174) = 2.2499,  $p < 0.05$ )

Rule number 2: Report the most sensible model. The most sensible model (m1\_2way) covers our three hypotheses. Thus, it models color, emotion, frequency and interaction between frequency manipulation and color as fixed effects. The random effects will model

pr. participant effect. The only significant variable is the interaction between color and frequency manipulation,  $T(2177) = 2.32, p < .05$ .

**Rule number 3:** Report the simplest model. The simplest model (beside the obvious null model) is a model only including color, emotion and frequency manipulation; no interactions. None of these variables in the simple model are significant.

**Rule number 4:** Report the most extensive and complete model. The most extensive and complete model is the full model with added no. as random slope and an added autoregressive component. There is one significant variable (cond\_blue:  $T\text{-value}(2174) = -2.7797, p < 0.01$ ) and two significant interactions (cond\_blue1:cond\_emo1:  $T\text{-value}(2174) = 2.3760, p < 0.05$  and cond\_blue1:freq:  $T\text{-value}(2174) = 1.9663, p < 0.05$ )

In conclusion, we would report the most sensible model: m1\_2way because it was our apriori model. It yields less significant results, but we will bite the sour apple.

#### **4.b.: Throughout part 3 of this exercise we made several models to choose from. What is the problem of this strategy? (This is analogous to the motivation for using family-wise-error corrected p-values in the SPM analysis)**

One problem might be that if you try enough models, you will probably find significant results. In a similar manner, doing enough t-tests in SPM (without correcting using family-wise error) will sooner or later yield a significant result. The alpha values are arbitrarily set, so that 1/20 tests/models will be significant by chance. We need to correct for this when doing many comparisons.

#### **4.c. Write a few lines, briefly stating the results of the experiment in relation to the hypotheses, using the model you decided upon in 4.a..**

We have decided upon using the most sensible model, i.e. the one using main effect for color, emotion, frequency and a 2-way interaction between frequency:color (m1\_2way), in accordance with our three hypotheses. According to this model, there was no effect of color, no effect of emotion; but there was an effect of infrequent stimuli in the two frequency manipulation groups. Thus, we reject H1 and H2; but we cannot reject H3.

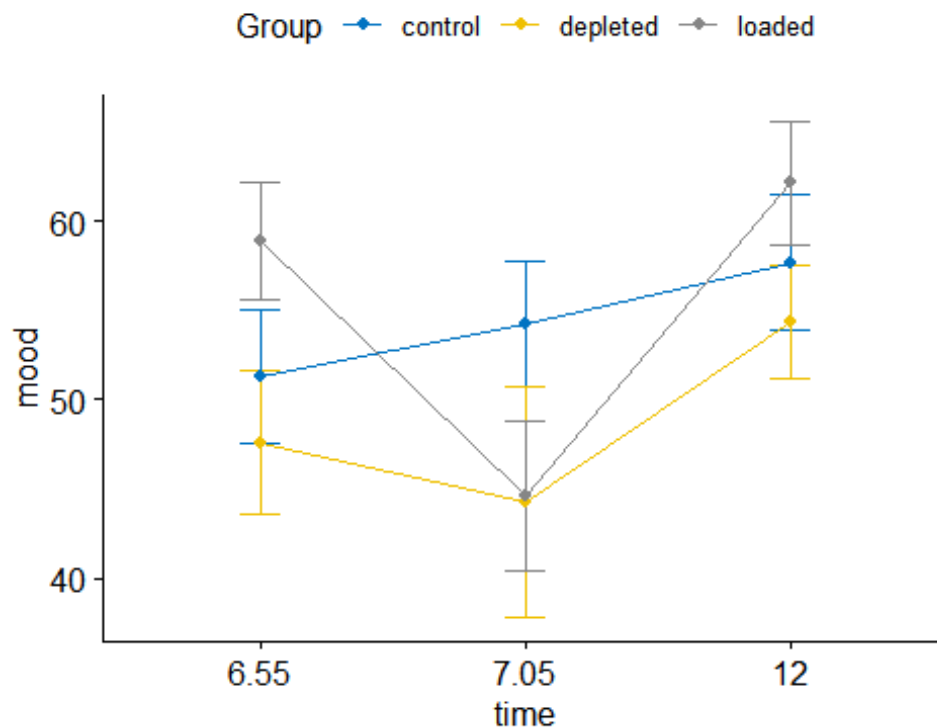
## **5. Interpretation task**

### **5.a. Find the data on Blackboard, load it and report figure and analysis using the code below.**

```
#Load data
trypt_long<-read.csv(file='trypt_long.csv',header=TRUE,sep=",")
trypt_long$ID<-as.factor(trypt_long$ID)
trypt_long$time<-as.factor(trypt_long$time)

#use ggline to make nice line plot. Install ggpubr, if you haven't got it
ggline(trypt_long, x = "time", y = "mood",col='Group',
add = c("mean_se", "dodge"), palette = "jco")
```





```
#Relevel to make the reference group "Loaded"
trypt_long$Group<-relevel(trypt_long$Group,'loaded')
#Relevel to make the reference time "7.05"
trypt_long$time<-relevel(trypt_long$time,'7.05')
#Make mixed effects model with Group and time as fixed effects and ID as random effect
trypt_model<-lmerTest::lmer(mood~Group*time+(1|ID), data = trypt_long)
#Get summary statistics
trypt_res<-summary(trypt_model)
trypt_res

## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: mood ~ Group * time + (1 | ID)
## Data: trypt_long
##
## REML criterion at convergence: 986.2
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.89017 -0.53496 -0.06987  0.55325  2.36443
##
## Random effects:
## Groups Name Variance Std.Dev.
## ID (Intercept) 112.7 10.62
## Residual 112.0 10.58
## Number of obs: 129, groups: ID, 43
```

```
##
## Fixed effects:
##
```

	Estimate	Std. Error	df	t value
## (Intercept)	44.5733	3.8709	79.8249	11.515
## Groupcontrol	9.6678	5.3108	79.8249	1.820
## Groupdepleted	-0.3279	5.9511	79.8249	-0.055
## time6.55	14.2867	3.8645	80.0000	3.697
## time12	17.5800	3.8645	80.0000	4.549
## Groupcontrol:time6.55	-17.2396	5.3021	80.0000	-3.251
## Groupdepleted:time6.55	-10.9594	5.9414	80.0000	-1.845
## Groupcontrol:time12	-14.1271	5.3021	80.0000	-2.664
## Groupdepleted:time12	-7.4709	5.9414	80.0000	-1.257

```
##
## Pr(>|t|)
## (Intercept) < 0.0000000000000002 ***
## Groupcontrol 0.072445 .
## Groupdepleted 0.956200
## time6.55 0.000398 ***
## time12 0.0000189 ***
## Groupcontrol:time6.55 0.001681 **
## Groupdepleted:time6.55 0.068798 .
## Groupcontrol:time12 0.009324 **
## Groupdepleted:time12 0.212252
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##      (Intr) Grpcnt Grpdpl tm6.55 time12 Grpc:6.55 Grpd:6.55 Grpc:12
## Groupcontrl -0.729
## Groupdepltd -0.650  0.474
## time6.55 -0.499  0.364  0.325
## time12 -0.499  0.364  0.325  0.500
## Grpcnt:6.55 0.364 -0.499 -0.237 -0.729 -0.364
## Grpdpl:6.55 0.325 -0.237 -0.499 -0.650 -0.325  0.474
## Grpcntrl:12 0.364 -0.499 -0.237 -0.364 -0.729  0.500  0.237
## Grpdpltd:12 0.325 -0.237 -0.499 -0.325 -0.650  0.237  0.500  0.474

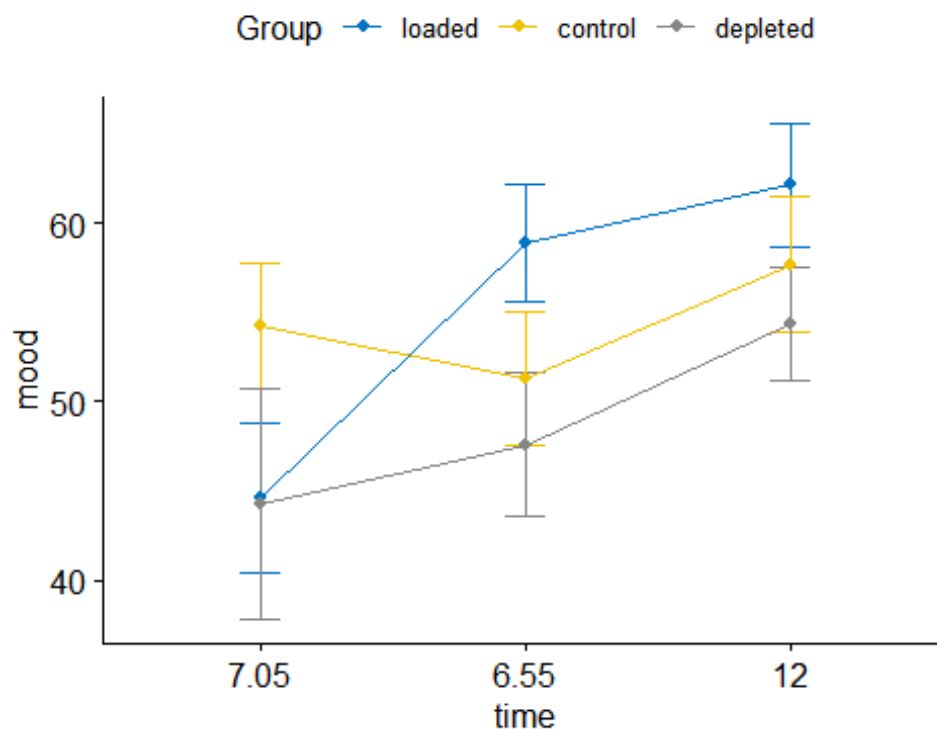
#Apply Bonferroni correction for multiple comparisons to p-values (9 tests)
# and round a bit (5 decimals)
trypt_res$coefficients2<-matrix(round(c(trypt_res$coefficients,trypt_res$coef
ficients[,5]*9),
digits=5),ncol=6)
#Add names to the new results matrix
colnames(trypt_res$coefficients2)<-c(colnames(trypt_res$coefficients),'p(bonf
)')
rownames(trypt_res$coefficients2)<-c(rownames(trypt_res$coefficients))
#Show us what you've got
trypt_res$coefficients2

##
```

	Estimate	Std. Error	df	t value	Pr(> t )
## (Intercept)	44.57333	3.87088	79.82486	11.51503	0.00000

```
## Groupcontrol          9.66784      5.31081 79.82486   1.82041   0.07244
## Groupdepleted        -0.32788      5.95114 79.82486  -0.05510   0.95620
## time6.55             14.28667      3.86452 80.00000   3.69688   0.00040
## time12                17.58000      3.86452 80.00000   4.54908   0.00002
## Groupcontrol:time6.55 -17.23961      5.30207 80.00000  -3.25148   0.00168
## Groupdepleted:time6.55 -10.95939      5.94136 80.00000  -1.84459   0.06880
## Groupcontrol:time12   -14.12706      5.30207 80.00000  -2.66444   0.00932
## Groupdepleted:time12   -7.47091      5.94136 80.00000  -1.25744   0.21225
##
## p(bonf)
## (Intercept)          0.00000
## Groupcontrol          0.65200
## Groupdepleted         8.60580
## time6.55              0.00359
## time12                0.00017
## Groupcontrol:time6.55  0.01513
## Groupdepleted:time6.55 0.61918
## Groupcontrol:time12    0.08392
## Groupdepleted:time12   1.91027
```

```
ggline(trypt_long, x = "time", y = "mood", col='Group',
add = c("mean_se", "dodge"), palette = "jco")
```



```
#Use library(emmeans) to get more comprehensible pairwise interactions (uncor
rected for multiple comparisons)
lsm = emmeans(trypt_model, ~Group*time)
contrast(lsm, interaction = "pairwise")
```

```
## Group_pairwise    time_pairwise estimate    SE df t.ratio p.value
## loaded - control  7.05 - 6.55    -17.240  5.30 80 -3.251  0.0017
## loaded - depleted 7.05 - 6.55    -10.959  5.94 80 -1.845  0.0688
## control - depleted 7.05 - 6.55     6.280  5.79 80  1.084  0.2815
## loaded - control  7.05 - 12      -14.127  5.30 80 -2.664  0.0093
## loaded - depleted 7.05 - 12      -7.471  5.94 80 -1.257  0.2123
## control - depleted 7.05 - 12      6.656  5.79 80  1.149  0.2539
## loaded - control  6.55 - 12       3.113  5.30 80  0.587  0.5588
## loaded - depleted 6.55 - 12       3.488  5.94 80  0.587  0.5588
## control - depleted 6.55 - 12      0.376  5.79 80  0.065  0.9484
##
## Degrees-of-freedom method: kenward-roger
```

### 5.b. Report and discuss the findings. What do they mean? How do they relate to the hypotheses?

Our intercept/baseline is the loaded group at 7.05 in the morning. There was a significant main effect of time at 6.55,  $T(80)=3.697$ ,  $p<.001$ . There was also a significant main effect of time at 12.00,  $T(80)=4.549$ ,  $p<.001$ . These were significant after using the bonferroni correction. However, there was not found a significant main effect of the group in which the participant was in.

Thus what seems to predict mood is not the presence of tryptophan, but the time of day at which mood was measured. The lowest mood for the tryptophan loaded group was reported at 7.05 just after eating the nasty powder. Before eating the nasty powder they were just fine; and at 12.00 they were also in a good mood.

When it comes to the interaction effects there was a significant effect of the interaction of the control group at 6.55 in the morning:  $\text{Groupcontrol:time6.55}$ ,  $T(80)=-3.251$ ,  $p<.01$ . There was also a significant effect of the interaction of the control group at 12:  $\text{Groupcontrol:time12}$ ,  $T(80)=-2.664$ ,  $p<.01$ .

However we do not know how to interpret these interactions. When looking at beta-estimates of the main effects, we can easily interpret them on the plot. But when looking at the beta-estimates of the interactions, we can't make sense of them using the plot. Please Help.